THE CLAIMS

Please enter the following claim amendment:

1. (Previously Presented) A system comprising:

a computer having a processing unit, a main memory, and a local bus;

a device coupled to the local bus, wherein the device occupies an I/O slot on the local bus and is accessible at a first set of addresses corresponding to a first communications port, and the device has a register set with an address assignment in the first set of addresses that differs from a standard address assignment of a register set for a UART corresponding to the I/O slot; and

a communications driver executed by the processing unit, the communications driver comprising a UART emulation which in response to an access targeted at a register set of a UART corresponding to the first communication port, converts the access as required for the register set and address assignment of the device.

2. (Original) The system of claim 1, wherein the local bus comprises an ISA bus.

3. (Previously Presented) The system of claim 1, wherein the device coupled to the local bus further comprises:

a comparator adapted for receiving a data signal from the local bus;

a pattern generator coupled to the comparator, wherein the pattern generator generates a signal for comparison with the data signal;

a counter operably coupled to the comparator, wherein the counter resets to an initial state following the comparator indicating the data signal is not equal to the pattern signal and advances toward a final state following the comparator indicating the data signal equals the pattern signal; and

a register coupled to the counter and adapted to receive a signal from the local bus, wherein in response to the counter reaching the final state, the register latches from the local bus a value which indicates the base address of the I/O slot occupied by the device.

4. **(Currently Amended)** A method for communicating between a computer and a device having an I/O interface which differs from the I/O interface of a UART, comprising:

coupling the I/O interface of the device to a local bus in the computer;

allocating in a memory of the computer, storage locations which correspond to registers of a UART;

transmitting a packet formatted for a UART via a communications driver including a UART emulation;

updating a value in the storage locations according to a value in the packet via the UART emulation; and

transmitting ~~information~~ the packet via the local bus between the I/O interface of the device and the allocated storage locations in the memory of the computer.

5. (Canceled).

6. (Previously Presented) The method of claim 4, wherein an I/O handler performs the step of said transmitting information by:

converting a value from the allocated storage to a converted value compatible with the I/O interface of the device; and

writing the converted value to a register in the device via the local bus.

7. (Previously Presented) The method of claim 4, wherein said transmitting information further comprises:

reading values from a register in the device via the local bus; and

updating the storage locations according to the value read.

8. (Previously Presented) The method of claim 7, further comprising transmitting from a communications driver to an application information from the storage locations.

9. (Original) The method of claim 4, further comprising:

executing on the computer an operating environment which allocates I/O slots on the local bus for UARTs; and

setting a base device address for the device to correspond to one of the I/O slots allocated by the operating environment for the UART.

10. (Previously Presented) The method of claim 9, wherein setting the base device address comprises:

sensing, by the device, of a data signal on the local bus;

comparing the data signal to a signal from a pattern generator in the device;

advancing a state indicator toward a final state in response to the data signal being equal to the signal from the pattern generator;

repeating the steps of sensing, comparing, and advancing until the state indicator reaches the final state; and

setting the base address of the device to a value indicated by a signal on the local bus in response to the state indicator reaching the final state.

11-16. (Canceled).

17. (Previously Presented) A host signal processing modem comprising:

a device adapted for connection to a local bus of a host computer, wherein the device occupies an I/O slot on the local bus and is accessible at a first set of addresses, the device having a register set with an address assignment in the first set of addresses that differs from a standard address assignment of a register set for a UART corresponding to the I/O slot; and

a communications driver executable by the host computer, the communication driver comprising a UART emulation, wherein in response to the host computer executing a procedure that targets an access at a register set of a UART, the UART emulation converts the access as required for accessing the register set and address assignment of the device.

18. (Previously Presented) The modem of claim 17, wherein the procedure that targets an access at the register set of a UART is part of an operating system that the host computer executes.

19. (Previously Presented) A communication driver executable by a host computer running an operating system that assigns a first port to a UART, the communication driver comprising:

a UART emulation that in response to a procedure requesting access to a register of a UART at a first port, instead accesses storage locations in a memory of the host computer; and

an I/O handler that transfers values between the storage locations and a register set of a non-standard device having an address assignment that differs from that of a UART, wherein the register set of the non-standard device physically occupies addresses corresponding to the first port.

20. (Canceled).

21. (Previously Presented) The communication driver of claim 19, further comprising modem software that implements a conversion between data and digital samples representing a signal in accordance with a communication protocol.

22. (Previously Presented) The communication driver of claim 19 wherein the address of a first storage location corresponds to a line control register, the address of and a second storage location corresponds to a line status register.

23. (Previously Presented) The host signal processing modem of claim 17 wherein the register set includes a line control register, a line status register and a modem control register.

24. (Previously Presented) A communication driver executable by a host computer running under an operating system, the communication driver comprising a software modem operable to allow a device with a non-standard input/output interface to

transparently communicate through the operating system with an application executing on the host computer.

25. (Previously Presented) The communication driver of claim 24, further comprising software that accesses storage locations in a memory of the host computer in response to a call requesting access to a register of a hardware UART.

26. (Previously Presented) The communication driver of claim 25, wherein the software modem converts between data and digital samples of waveforms in accordance with a modem protocol.

27. (Previously Presented) The communication driver of claim 26, further comprising an I/O handler that transfers values between storage locations in the memory of the host computer and a register set of a non-UART chip in a peripheral device of a host computer.

28. (Previously Presented) The communication driver of claim 27, wherein the peripheral device further comprises an analog-to-digital converter and a digital-to-analog converter.

29. (Canceled).

30. (Previously Presented) A system comprising:

a device comprising an analog to digital converter couplable to a communication medium to receive therefrom an analog communications signal; and

a computer comprising a processing unit coupled to the device, to receive therefrom a plurality of sampled digital values, the processing unit being programmed with a software modem to determine data received, based on a waveform represented by the sample digital values, wherein the processing unit is programmed with an operating system for supporting a plurality of applications, at least one of the applications communicating with the software modem in the same manner as with a hardware modem.

31. (Previously Presented) The system of Claim 30 wherein:
the device generates interrupts; and
the software modem reads a set of sampled digital values from the analog to digital converter in response to an interrupt.

32. (Previously Presented) The system of Claim 30 wherein:
the device further comprises a digital to analog converter coupled to the communication medium to transmit thereto an analog signal; and
the software modem generates a series of digital values sent to the digital to analog converter for transmission as an analog signal on the communication medium, the analog signal providing a carrier signal and data values formatted according to a standard modem protocol.

33. (Previously Presented) A method comprising:
converting an analog communications signal received from a device with a non-standard input/output interface into a series of sampled digital values, wherein said act of converting is performed in an analog to digital converter;
determining data received based on a waveform represented by the sampled digital values, and based on a modem protocol, wherein said determining is performed in a processing unit coupled to the analog to digital converter by a local bus of a computer, the processing unit running under an operating system, the sampled digital values being transferred from the analog to digital converter to the processing unit by the local bus; and
providing the received data through the operating system to an application executing on the processing unit, thereby allowing the device with the non-standard input/output interface to transparently communicate with the application.

34. (Previously Presented) The method of Claim 33, wherein:
the analog to digital converter is a portion of a device that does not comprise a standard UART, and the method further comprises the processing unit determining if a non-standard UART device is present.

35. (Previously Presented) The method of Claim 33 further comprising:
generating a series of digital values, in said processing unit; and
transmitting an analog signal based on the series of digital values, in a digital to analog converter, wherein said digital to analog converter is coupled to the processing unit by the bus.

36-37. (Canceled).

38. (Previously Presented) A computer comprising:
a processing unit and memory programmed with a driver, said driver comprising (a) a software UART coupled to an operating system, (b) a software modem coupled to the software UART, and (c) an I/O handler coupled to the software modem; and
a device that does not comprise a UART, the device being coupled to the processing unit by a local bus, wherein the device comprises an analog to digital converter that generates sample digital values, and the device transfers the sampled digital values via the local bus to the software modem.

39. (Previously Presented) The computer of Claim 38 wherein said device is hereinafter "first device" and said driver is hereinafter "first driver", the computer further comprising
a second device comprising a UART, the second device being coupled to the processing unit by the local bus; and
a second driver comprising routines for accessing the second device.

40. (Previously Presented) The computer of Claim 39 wherein:
said first device is coupled to an I/O slot corresponding to a first COM port; and
said second device is coupled to an I/O slot corresponding to a second COM port.

41. (Previously Presented) The computer of Claim 40 wherein:
said first device occupies up to eight addresses on the local bus; and

said second device occupies eight addresses on the local bus.

42. (Previously Presented) The computer of Claim 38 wherein:
the memory is further programmed with routines for accessing another device that comprises a UART.

43. (Previously Presented) A computer comprising:
an analog to digital converter couplable to a device with a non-standard input/output interface to receive therefrom an analog communications signal and coupled to a local bus to transmit thereto a series of sampled digital values; and
a processor coupled to the local bus, the processor running under an operating system and programmed to:
determine data received based on a waveform represented by the sampled digital values and based on a modem protocol; and
provide the received data through the operating system of the computer to an application executing on the processor, thereby allowing the device with the non-standard input/output interface to transparently communicate with the application.

44. (Previously Presented) The computer of Claim 43 wherein:
said processor is programmed to transmit a series of digital values on the local bus; and
the computer further comprises a digital to analog converter coupled to the local bus to receive therefrom the series of digital values and couplable to the communication medium to transmit thereto an analog signal based on the series of digital values.

45. (Previously Presented) The computer of Claim 44, wherein:
the analog to digital converter and the digital to analog converter are portions of a first device that does not comprise a standard UART;

the computer further comprises a second device that comprises a standard UART; and

said processor is programmed to:

access the first device through the operating system and a software UART; and

access the second device through the operating system and a standard COM driver.

Please add the following new claims:

46. (Newly Presented) A protocol translation apparatus comprising:

a first device communicating using a first protocol;

a second device in communication with the first device via a serial input/output interface for communicating using a second protocol, wherein the second device includes a processor, a local bus, and memory and wherein the second protocol emulates at least a portion of a UART protocol; and

a translation means in communication with the first device and the processor of the second device, wherein the translation means designates portions of the memory to correspond to the output portion of the interface and another portion of the memory to correspond to the input portion of the interface, such that communication between the first and second device is routed through the memory by the translation means.

47. (Newly Presented) The apparatus of claim 46, wherein the first device occupies up to eight addresses on the local bus and the second device occupies eight addresses on the local bus.

48. (Newly Presented) The apparatus of claim 46, wherein the memory is further programmed with routines for accessing another device that comprises a UART.

49. (Newly Presented) The apparatus of claim 46, wherein the translation means comprises:

a processing unit and a driver, wherein the driver comprises:

a software UART;

a software modem in communication with the software UART, and

an input/output handler in communication with the software modem.

50. (Newly Presented) A system comprising:

a computer having a processing unit, a main memory, and a bus for communicating with devices connected thereto, the bus having at least one I/O slot;

a device coupled to the bus and operating over a substantially serial interface, wherein the device occupies an I/O slot on the local bus and is accessible at a first set of addresses corresponding to a first communications port, the device having a register set with an address assignment in the first set of addresses that differs from a standard address assignment of a register set for a UART corresponding to the I/O slot; and

a communications driver executed by the processing unit, the communications driver comprising an emulation of at least a portion of a UART which in response to an access targeted at a register set of a UART corresponding to the first communication port, converts the access as required for the register set and address assignment of the device.

51. (Newly Presented) The system of claim 51, wherein the UART emulation by the communications driver, in response to a procedure requesting access to a register of a UART at a first port of the interface, instead accesses the first memory portion.